

SPECIFICATION
TITLE

METHOD FOR THE REMOVAL OF ATM CELLS FROM AN ATM
COMMUNICATIONS DEVICE

Field of the Invention

The invention is directed to a method for the removal of ATM cells from an ATM communications device in which a respective plurality of ATM cells are allocated to a common frame.

Description of the Related Art

In a traditional packet communication system, a packet has a comparatively great and variable length. A system for the transmission of information in packets having fixed, predetermined lengths is referred to as an ATM (asynchronous transfer mode) system. Voice, video and data signals can be processed and transmitted in the same way with such a system. The individual packets are usually called cells. A cell header, whose information enables a switching or, respectively, allocation of the respective cell is respectively contained in the cells. A high-speed and broadband transmission with a transmission rate of more than 150 Mb/s is possible in ATM communication devices, particularly communication network devices.

In particular, the ATM cells have a length of 53 bytes for a broadband ISDN (Integrated Services Digital Network). The publication by M. DePrycker, "Asynchronous Transfer Mode", 2nd Edition, London, Horwood, 1993, is referenced for further details about the structure of ATM cells.

One problem given ATM communication devices is the height of the transmission rate on a transmission link of the device when a back-up of ATM cells has formed at the transmission link. The problem is described in greater detail in the publication in the IEEE Journal on Selected Areas in Communications, Vol. 13, No. 4, May 1995, pages 633 through 641, "Dynamics of TCP Traffic over ATM Networks" by Allyn Romanow and Sally Floyd (referred to as IEEE 95 below). The article is concerned with ATM systems wherein a respective plurality of ATM cells are allocated to a common frame. When, for example, a cell of such a frame has been lost or damaged, it is undesirable that the further cells of the same frame are transmitted via a transmission link of an ATM device since the complete information of the frame would no longer arrive at the end of the transmission link. The ATM system would be unnecessarily burdened.

the publication

²

A

A

the Publication

SECRET

A
A
A

privileged.

A

IEEE 95 discloses another

A
A

added to the waiting list.

SUMMARY OF THE INVENTION

Providing
ing a method for

Ans a'

This object is achieved by a method ~~having the features of claim 1.~~
~~Developments are the subject matter of the dependent claims.~~

What is understood by the term "waiting list" in this specification is any administration unit for the administration of a plurality of ATM cells in which a one-
 5 dimensional logical chaining of the plurality of ATM cells is formed or can be produced. Waiting lists that are organized according to the FIFO principle are particularly included here.

A

In the inventive method for the removal of ATM cells, all ATM cells of a frame whose first ATM cell is located in the waiting list are removed from a waiting
 10 list for the administration of a sequence of ATM cells. The greatest possible space in the waiting list is thus created given removal of the ATM cells of the frame. Over and above this, it is possible to remove all ATM cells of the frame simultaneously or immediately successively from the waiting list, so that the cells can be removed in the shortest possible time. In particular, it is not necessary to thereby wait for the arrival
 15 of ATM cells at the waiting list. The method can also be applied given a plurality of conditions of the waiting list, namely whenever the first ATM cell of a frame is in the waiting list. Given developments of the method, the application of the method can, however, be made dependent on meeting additional conditions.

When, for example, the waiting list is realized with a pointer chain, the
 20 deletion or, respectively, removal of the ATM cells can be realized simply by parting the pointer chain and by releasing the corresponding memory space in a data store. As known, the pointer chain can be produced and administered with a computer program or with hardware, particularly upon utilization of the information at defined locations in hardware data store areas for accepting ATM cell information.

25 In a development of the method, the frame whose ATM cells are removed is the frame that begins farthest toward the back in the waiting list. Upon removal of the ATM cells of the frame, it is particularly the first ATM cell thereof that is identified, and this ATM cell as well - when present - all following ATM cells of the frame located in the waiting list are removed.

Preferably, following ATM cells of the frame are removed at or following arrival at the waiting list up to and including the last ATM cell of the frame. What this prevents is that the ATM cells of the frame arriving later unnecessarily burden the ATM communications device. The removal of the following ATM cells is the same
 5 as the removal of ATM cells according to EPD insofar as the removal of the individual cells is triggered by their arrival at the waiting list.

A frame start identifier is preferably stored that references the ATM cell in the waiting list immediately preceding the first ATM cell of the frame, and the frame start identifier is called before the removal of the ATM cell or, respectively, of the
 10 ATM cells of the frame. This procedure has the advantage that the information usually present in ATM systems regarding which cell is the last cell of a frame can be utilized. This information is usually present in the cell header of the last cell of the frame, namely in what is referred to as the AAU bit in the cell type field (payload-type field) of the cell header as a rule.

In particular, the presence of this information is respectively checked at or before the adding of a newly arrived ATM cell to the end of the waiting list. As
 15 warranted, a value is then stored as ^a frame start identifier that identifies this ATM cell, so that the ATM cells of the appertaining frame cannot be removed from the waiting list since - at least in this status of the waiting list - no first ATM cell of a following
 20 frame is present in the waiting list after the last ATM cell of the frame that has just arrived. As soon as such a first ATM cell of a following frame has arrived, a removal of ATM cells of the following frame is possible.

In particular, the above-described measure serves the purpose of protecting individual ATM cells not allocated to any frame, particularly OAM cells (operation,
 25 administration, maintenance) or RL cells (resource management) in a development of the method. OAM cells generally serve for administration and maintenance; RM cells serve for flow control. Such individual cells should often not be removed from the ATM communications device. When such an individual ATM cell immediately precedes the first ATM cell of the frame that is the only frame beginning in the
 30 waiting list or is the frame that begins farthest toward the back in the waiting list, a

0962375-090800

A

value that references this individual ATM cell is therefore preferably stored as the frame start identifier. Consequently, this individual cell is protected from being removed because, in this development of the method, only following cells in the waiting list can be removed.

5 When a last ATM cell of another frame immediately precedes the first ATM cell of the frame that is the frame beginning farthest toward the back in the waiting list, the frame start identifier preferably references this ATM cell.

Another possibility of protecting individual ATM cells not allocated to any frame is realized in a development. In this development, a predetermined block value is stored instead of the start identifier when the first ATM cell of the frame whose ATM cells come into consideration for removal from the waiting list is followed by such an individual ATM cell. The block value is preferably stored upon arrival of the individual ATM cell at the waiting list and/or when this cell is added to the waiting list. The cell is thus immediately protected after it arrives or, respectively, is added.

15 **BRIEF DESCRIPTION OF THE DRAWINGS**
The invention is now described in greater detail on the basis of exemplary embodiments. However, it is not limited to these exemplary embodiments. ~~The individual Figures of the drawing show:~~

Fig. 1 ^{is a diagram of} a waiting list for the administration of a sequence of ATM cells; and
20 Fig. 2 ^{is a diagram of} the procedure of removing ATM cells proceeding from the status of a waiting list shown in Fig. 1.
DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 shows a waiting list 1 in which ATM cells 2, 3, 4, 5, 6 are arranged in a specific sequence. The ATM cells are thereby partially allocated to two different frames 8 ^{and} 9, whereby further ATM cells of the frame 8 have already left the waiting list 1 in the direction of the arrow toward the right, and further ATM cells of the frame 9 have not yet arrived at the waiting list 1 (coming from the left). The first waiting list cell 5 is therefore not the first cell of the frame 8. The last frame cell 3 of the frame 8, which carries a corresponding frame end identifier in its cell header, is in the waiting list 1. This last frame cell ⁴3 is immediately followed by an OAM cell that
30 is an individual cell not allocated to any frame. The OAM cell 4 is immediately

[illegible]

and Variables, particularly pointers, are provided for marking specific cells 4, 5, 6 in the waiting list 1, particularly with a computer program for the administration of the cells in the waiting list 1. The variable P__first__cell thereby references the first waiting list cell 5 of the waiting list 1. When the waiting list 1 is empty, then a predetermined value is stored in the variable, referred to as “invalid” below, which means that no valid entry is present. A value that references the last ATM cell that is a last frame cell in the waiting list 1 is stored in the variable P__end__of__frame.

When such a last frame cell is followed, for example as in Figure 1, by the ATM cell 3, an individual cell not allocated to any frame, the OAM cell 4 in the example of Figure 1, then the identifier of the last cell, i.e. the individual cell located farthest toward the back of the waiting list 1, is stored in the P_end_of_frame. In the example of Figure 1, only one such cell is present, so that P_end_of_frame contains the identifier of the OAM cell 4.

When space is then to be created in the ATM system, particularly for ATM cells having a higher priority, then, as shown in Figure 2, all cells of the frame 9 that are already in the waiting list 1 are initially removed from the waiting list. To that end, the value of the cell that is already entered in the variable P_end of __frame is preferably entered in the variable P__last__cell. In the example of Figure 2, this is the OAM cell 4. Further, the value TRUE is preferably entered in a variable LPD_flag. In order to enable a query as to whether the procedure of the removal of ATM cells is activated. LPD is the abbreviation for last packet discard, which means that the last frame in the waiting list is removed.

25 Following ATM cells 7 of the frame 9 are then removed at or after the
arrival at the waiting list 1. The status shown in the upper part of Figure 2 has thus
been reached. The further, following ATM cells of the frame 9 up to the frame end
10, i.e. up to the last ATM cell 3 of the frame 9, are removed at or following the
arrival at the waiting list 1. The status of the waiting list 1 shown in the lower part of
30 Figure 2 has thus been reached.

An exemplary embodiment of the invention is now described below on the basis of parts of a computer program for administering a sequence of ATM cells in a waiting list. Such computer programs are also employed in the known methods of early packet discard (EPD) and partial packet discard (PPD). Routines of the computer program described below, however, partially differs significantly from the known computer programs.

The following assumptions are made: ATM cells arrive at a waiting list. Some of these cells as well as cells that are already classified in the waiting list are to be removed. The remaining cells leave the waiting list in the meantime or later. The ATM cells are at least partially organized in frames, i.e. successive ATM cells from a first frame cell up to a last frame cell belong to a common frame. No frame cells of a different frame are located between the first and the last frame cell. However, individual ATM cells not allocated to any frame can be arranged between the first frame cell and the last frame cell. That stated above applies both to the sequence in a waiting list as well as to the sequence of the transmission on a transmission link of an ATM communications device. The last ATM cell of a respective frame can be unambiguously identified. An unambiguous, one-dimensional chaining of the cells is produced in the waiting list for the administration of the ATM cells in the waiting list. The sequence is thus unambiguously defined. For locating specific cells in the waiting list, however, it would last too long if the search were always begun at the beginning or end of the waiting list and the cells had to be checked cell-by-cell in the waiting list. Following cells can therefore be directly located by storing an identifier in a variable:

- the first cell in the waiting list (variable: P_first_cell)
- the last cell in the waiting list (variable: P_last_cell)
- the last cell in the waiting list that is a last frame cell or that is an individual cell not allocated to any frame and that is arranged between two frames (variable: P_end_of_frame).

The removal of ATM cells according to the LPD method is only implemented when the last frame cell of the waiting list is not the cell whose

identifier is deposited in `P_end_of_frame` and when a valid cell identifier is entered in the variable `P_end_of_frame`, i.e. when a last frame cell or an individual cell following thereupon is still in the waiting list.

A plurality of waiting lists can be present in an ATM communications device, these being respectively administered according to the method described below. In this case, each waiting list has its own individual identifier, and variables for storing the aforementioned cells are present in each waiting list. For the sake of simplicity, it is assumed for the following program parts that only one waiting list is present.

First, individual operations shall be presented that can be implemented at the cells. It is assumed that each of the cells has an unambiguous identifier that is referenced `P_cell`. The operations are:

- `next_cell (P_cell)` returns the identifier of the immediately following cell in the waiting list.
- 15 - `end_of_frame (P_cell)` returns the value TRUE when `P_cell` references a last frame cell and otherwise returns the value FALSE.
- `exclude_cell (P_cell)` returns the value TRUE for cells that are not to be removed, for example OAM cells
- 20 - `discard_cell (P_cell)` removes the cells with the identifier `P_cell`
- `decide_cell (P_cell)` determines on the basis of criteria that are not explained in detail here whether specific operations or procedures, particularly `discard_cell` or `append_cell` (see below) are to be carried out at the cell having the identifier `P_cell`.
- 25

The following procedures or, respectively, functions (called procedures below) are explained in greater detail:

- *arrive_cell (P_cell)* implements various operations at the cell having the identifier P_cell upon arrival at the waiting list.
- *queue_empty* returns the value TRUE when the waiting list is empty and otherwise returns the value FALSE.
- *append_cell (P_cell)* attaches the cell having the identifier P_cell to the end of the waiting list and implements various operations
- 10 - *extract_cell* serves for the removal of a cell at the start of the waiting list, particularly for the transmission of this cell onto a transmission link
- *remove_last_frame* removes all cells of the last frame of the waiting list from the waiting list, if possible.

In order to enable an inquiry as to whether the removal of cells from the ATM communications device according to the method LPD is active, a boolean variable LPD_flag is provided.

20 In the initialization of the program, i.e. when the waiting list is empty, the three cell identifier variables P_first_cell, P_last_cell and P_end_of_frame are set to the value "invalid", and the variable LPD_flag is set to the value FALSE.

Cells that are not ordinary data cells and that do not belong to a frame can arrive at the waiting list. These individual cells, for example OAM cells or RM cells, can be excluded from removal from the ATM communications device or can be not excluded therefrom. Criteria that are not explained in greater detail here are available for this purpose in the ATM communication system. When one of these cells that is not to be removed is arranged within a first and a last frame cell, then the removal of the cells of the frame is not implemented according to the method LPD.

30 Procedure *arrive_cell* works according to the following algorithm in the exemplary program:

5
10

15
20

25

30

```

P_last_cell = P_cell
ELSE IF exclude_cell (P_cell)
    THEN IF P_end_of_frame=(P_last_cell)
        /*both are valid implicitly*/
5      THEN P_end_of_frame = P_cell
        ELSE P_end_of_frame = invalid
    next_cell (P_last_cell) = P_cell
    P_last_cell = P_cell
    IF end_of_frame (P_cell)
10    /*cell with identifier P_cell is the last cell of the frame*/
    THEN P_end_of_frame = P_cell

```

In the procedure *append_cell* (*P_cell*), a check is first carried out to see whether the cell having the identifier *P_cell* is to be removed in any case. Potentially, the procedure *discard_cell* (*P_cell*) is called. Otherwise, a check is carried out to see whether the waiting list is empty. If it is, the identifier of the cell is entered in the variables *P_first_cell* and *P_last_cell*. When the waiting list was not empty, a check is carried out to see whether the cell is to be protected against removal in any case because, for example, it is an OAM cell. When this cell is to be protected in every case, either the identifier of the cell is entered in the variable *P_end_of_frame* (when the identifier of a last frame cell was previously entered in the variable) or the value "invalid" is otherwise entered. When the cell with the identifier *P_cell* is itself a last frame cell, its identifier is entered in the variable *P_end_of_frame*. In order to attach the cell to the waiting list, a pointer connection to the attached cell is produced regardless of the previous status of the waiting list and regardless of the nature of the cell to be inserted, and the identifier of the cell is entered in the variable *P_last_cell*.

The procedure *extract_cell* is described by the following algorithm in the exemplary program:

```

IF      NOT (queue__empty)
THEN IF P_first_cell = P_end_of_frame
30    THEN P_end_of_frame = invalid

```

$$P_first_cell = next_cell \quad (P_first_cell)$$

5 taken - when the waiting list is not empty - for further processing, particularly transmission, from the waiting list. A check is thereby carried out to see whether the first cell is a matter of a last frame cell or, respectively, a matter of an individual cell entered in the variable P_end_of__frame. In this case, the value "invalid" is entered in the variable P_end_of__frame, since, following the removal of the first cell, a
10 corresponding cell is then no longer located in the waiting list. In particular, a last frame cell is then no longer in the waiting list. A removal of cells from the waiting list is then not possible until a last frame cell and a frame cell of a following frame following thereupon have been attached to the waiting list. The first waiting list cell is removed and the identifier of the next-successive cell in the waiting list is entered
15 in the variable P_first_cell.

The procedure *remove_last_frame* is described by the following algorithm in the exemplary program:

```

    IF      NOT
            (queue_empty OR
20      (P_end_of_frame = invalid) OR
            (P_end_of_frame = P_last_cell))
            /*last frame can be removed*/
    THEN    P_last_cell = P_end_of_frame
            LPD_flag  = TRUE
25      retrieve storage starting at cell with identifier
            next_cell (P_last_cell)

```

is the value “invalid” entered in the variable P_end_of_frame?

is the same identifier entered in the variable P_end_of_frame and in the

list. New values are only entered in two variables, namely P_last_cell and LPD_flag, and the corresponding memory space is released. The release of the memory space can thereby be particularly implemented step-by-step when free calculating time is available. The system is thus available for further cell operations within the shortest possible time.

5

ms A2

0976